L Number	Hits	Search Text	DB	Time stamp
1	112	debug$4 same (trouble adj shoot$3)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:24
8	22	(port$1 or interface$1) same (debug$4 same (trouble adj shoot$3))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:24
15	2	JTAG same ((port$1 or interface$1) same (debug$4 same (trouble adj shoot$3)))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:24
29	0	(debug$4 same (trouble adj shoot$3)) and (hardware near2 state$1 near3 static)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:15
22	4	hardware near2 state$1 near3 static	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:19
36	587	(stop or stopp$3 or idle$1) near3 ((system or bus) adj clock$1)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:22
43	0	(debug$4 same (trouble adj shoot$3)) same ((stop or stopp$3 or idle$1) near3 ((system or bus) adj clock$1))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:22
50	1	(debug$4 same (trouble adj shoot$3)) and ((stop or stopp$3 or idle$1) near3 ((system or bus) adj clock$1))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:22
57	25967	debug$4 or (trouble adj shoot$3)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:24
64	5142	(port$1 or interface$1) same (debug$4 or (trouble adj shoot$3))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:24
71	336	JTAG same ((port$1 or interface$1) same (debug$4 or (trouble adj shoot$3)))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:24
78	3	JTAG same ((stop or stopp$3 or idle$1) near3 ((system or bus) adj clock$1))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:24

L Number	Hits	Search Text	DB	Time stamp
1	112	debug$4 same (trouble adj shoot$3)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:02
8	22	(port$1 or interface$1) same (debug$4 same (trouble adj shoot$3))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:14
15	2	JTAG same ((port$1 or interface$1) same (debug$4 same (trouble adj shoot$3)))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:15
22	4	hardware near2 state$1 near3 static	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:15
29	0	(debug$4 same (trouble adj shoot$3)) and (hardware near2 state$1 near3 static)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2003/06/29 20:15

US-PAT-NO: 6202172

DOCUMENT-IDENTIFIER: US 6202172 B1

TITLE: Smart debug interface circuit

DATE-ISSUED: March 13, 2001

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Ponte; Christian	Colle sur Loup	N/A	N/A	FR

US-CL-CURRENT: 714/31, 712/227

ABSTRACT:

The present invention comprises a smart debug interface circuit for the diagnostic testing and debugging of a software application for a programmable digital processor system. The smart debug interface circuit of the present invention includes an instruction register for coupling to an instruction bus of a programmable digital processor. The instruction register is adapted to drive instructions onto the instruction bus. The instruction register couples to the instruction bus in a parallel manner. The smart debug interface circuit of the present invention includes a data register for coupling to a data bus of the programmable digital processor. The data register is adapted to read data from the data bus and couples to the data bus in a parallel manner. The instruction register and data register are each coupled to an interface port. The interface port couples the smart debug interface circuit to a host computer system. A control logic circuit is also included in the smart debug interface circuit of the present invention. The control logic circuit is coupled to the instruction register, the data register, and the interface port. The control logic circuit interfaces a debugging program on the host computer system to the programmable digital processor. Additionally, the control logic circuit interfaces the debugging program with the programmable digital processor without imposing boundary scan bus delay on the instruction bus or the data bus.

3 Claims, 5 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 5

---------- KWIC ---------

Brief Summary Text - BSTX (17):

In this manner, the smart **debug interface** circuit of the present invention provides a diagnostic **trouble shooting and debugging** solution which provides the benefits of conventional test **interfaces** while avoiding their associated adverse impacts on the programmable digital processor system under test. The system of the present invention is fully compatible with the industry standard JTAG **interface**. The system of the present invention does not increase the size of the programmable digital processor integrated circuit by including numerous serial scan cells. In addition, the system of the present invention does not impose a boundary scan delay by breaking the programmable digital processor busses with boundary scan cells. As a further advantage, the smart **debug interface** circuit of the present invention does not increase the amount of

memory required to store the application software of the programmable digital processor system.

Detailed Description Text - DETX (3):

The present invention comprises a smart **debug interface** circuit for performing diagnostic testing and **debugging** of a programmable digital processor system. The smart **debug interface** circuit of the present invention is a JTAG compliant **interface for debugging,** diagnosing, and **trouble shooting** devices incorporating programmable digital processors. The present invention includes a **debugging** program running on an external host computer system. The **debugging** program is **interfaced** with the programmable digital processor device via the smart **debug interface** circuit, allowing the efficient and accurate **trouble shooting** of the programmable digital processor device. Additionally, the control logic circuit **interfaces the debugging** program with the programmable digital processor without imposing boundary scan bus delay on the instruction bus or the data bus of the programmable digital processor device. In this manner, the smart **debug interface** circuit of the present invention provides a diagnostic **trouble shooting and debugging** solution providing the benefits of conventional test **interfaces** while avoiding their associated adverse impacts on the programmable digital processor system under test. The system of the present invention is fully compatible with the industry standard JTAG **interface,** does not increase the size of the programmable digital processor integrated circuit by including numerous serial scan cells, and does not impose a boundary scan delay from breaking the programmable digital processor busses. The present invention and its benefits are described in greater detail below.

Detailed Description Text - DETX (30):

Thus, the SDI of the present invention provides a diagnostic **trouble shooting and debugging** solution which provides the benefits of conventional test **interfaces** while avoiding their associated adverse impacts on the programmable digital processor system under test. The system of the present invention is fully controllable and compatible with the industry standard JTAG **interface**. The system of the present invention does not increase the size of the programmable digital pirocessor integrated circuit by including numerous serial scan cells. In addition, the system of the present invention does not impose a boundary scan delay by breaking the programmable digital processor busses with boundary scan cells. As a further advantage, the smart **debug interface** circuit of the present invention does not increase the amount of memory required to store the application software of the programmable digital processor system.

US-PAT-NO: 5915083

DOCUMENT-IDENTIFIER: US 5915083 A
See image for Certificate of Correction

TITLE: Smart debug interface circuit for efficiently for debugging a software application for a programmable digital processor device

DATE-ISSUED: June 22, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Ponte; Christian	La Colle sur Loup	N/A	N/A	FR

US-CL-CURRENT: 714/30

ABSTRACT:

The present invention comprises a smart debug interface circuit for the diagnostic testing and debugging of a software application for a programmable digital processor system. The smart debug interface circuit of the present invention includes an instruction register for coupling to an instruction bus of a programmable digital processor. The instruction register is adapted to drive instructions onto the instruction bus. The instruction register couples to the instruction bus in a parallel manner. The smart debug interface circuit of the present invention includes a data register for coupling to a data bus of the programmable digital processor. The data register is adapted to read data from the data bus and couples to the data bus in a parallel manner. The instruction register and data register are each coupled to an interface port. The interface port couples the smart debug interface circuit to a host computer system. A control logic circuit is also included in the smart debug interface circuit of the present invention. The control logic circuit is coupled to the instruction register, the data register, and the interface port. The control logic circuit interfaces a debugging program on the host computer system to the programmable digital processor. Additionally, the control logic circuit interfaces the debugging program with the programmable digital processor without imposing boundary scan bus delay on the instruction bus or the data bus.

17 Claims, 5 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 5

---------- KWIC ---------

Brief Summary Text - BSTX (17):

In this manner, the smart **debug interface** circuit of the present invention provides a diagnostic **trouble shooting and debugging** solution which provides the benefits of conventional test **interfaces** while avoiding their associated adverse impacts on the programmable digital processor system under test. The system of the present invention is fully compatible with the industry standard JTAG **interface**. The system of the present invention does not increase the size of the programmable digital processor integrated circuit by including numerous serial scan cells. In addition, the system of the present invention does not

impose a boundary scan delay by breaking the programmable digital processor busses with boundary scan cells. As a further advantage, the smart **debug interface** circuit of the present invention does not increase the amount of memory required to store the application software of the programmable digital processor system.

Detailed Description Text - DETX (3):

The present invention comprises a smart **debug interface** circuit for performing diagnostic testing and **debugging** of a programmable digital processor system. The smart **debug interface** circuit of the present invention is a JTAG compliant **interface for debugging,** diagnosing, and **trouble shooting** devices incorporating programmable digital processors. The present invention includes a **debugging** program running on an external host computer system. The **debugging** program is **interfaced** with the programmable digital processor device via the smart **debug interface** circuit, allowing the efficient and accurate **trouble shooting** of the programmable digital processor device. Additionally, the control logic circuit **interfaces the debugging** program with the programmable digital processor without imposing boundary scan bus delay on the instruction bus or the data bus of the programmable digital processor device. In this manner, the smart **debug interface** circuit of the present invention provides a diagnostic **trouble shooting and debugging** solution providing the benefits of conventional test **interfaces** while avoiding their associated adverse impacts on the programmable digital processor system under test. The system of the present invention is fully compatible with the industry standard JTAG **interface,** does not increase the size of the programmable digital processor integrated circuit by including numerous serial scan cells, and does not impose a boundary scan delay from breaking the programmable digital processor busses. The present invention and its benefits are described in greater detail below.

Detailed Description Text - DETX (30):

Thus, the SDI of the present invention provides a diagnostic **trouble shooting and debugging** solution which provides the benefits of conventional test **interfaces** while avoiding their associated adverse impacts on the programmable digital processor system under test. The system of the present invention is fully controllable and compatible with the industry standard JTAG **interface**. The system of the present invention does not increase the size of the programmable digital processor integrated circuit by including numerous serial scan cells. In addition, the system of the present invention does not impose a boundary scan delay by breaking the programmable digital processor busses with boundary scan cells. As a further advantage, the smart **debug interface** circuit of the present invention does not increase the amount of memory required to store the application software of the programmable digital processor system.

US-PAT-NO: 5896415

DOCUMENT-IDENTIFIER: US 5896415 A

TITLE: Device for out-of-band transmission on computer peripheral

DATE-ISSUED: April 20, 1999

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Owens; Craig	Belmont	CA	N/A	N/A
Griesing; John	Saratoga	CA	N/A	N/A

US-CL-CURRENT: 375/224, 375/222 , 375/257 , 375/377 , 379/1.03

ABSTRACT:

A peripheral device for use with a computer. The peripheral device includes a data path, a control circuit coupled to the data path for controlling data communication and a port. The port includes a connector used by the data path. The connector includes a first set of pins that are used by the data path and a second set of pins that are not used by the data path. The first set of pins are coupled to the data path for data communication from the data path to the computer, and the second set of pins coupled to the control circuit for providing access to the control circuit from the port.

14 Claims, 5 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 5

---------- KWIC ---------

Brief Summary Text - BSTX (5):

Configuration and maintenance of electronic computer peripheral equipment is critical. When a user has purchased a peripheral device, the device may need to be set-up. Later, during the use of the device, the device may require **trouble shooting, debugging,** or diagnosis. Standard **interfaces** provided on devices may not provide sufficient access to circuitry within the devices to allow for performance of diagnosis, assessment of status, or **debugging**. However, physically opening the device in order to gain access to the circuitry in the device may be time consuming, inconvenient, or destructive to the device.

US-PAT-NO: 5473754

DOCUMENT-IDENTIFIER: US 5473754 A

TITLE: Branch decision encoding scheme

DATE-ISSUED: December 5, 1995

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Folwell; Dale E.	Placentia	CA	N/A	N/A
Clark; Ricke W.	Irvine	CA	N/A	N/A
Harenberg; Donald D.	Placentia	CA	N/A	N/A

US-CL-CURRENT: 714/45, 712/227

ABSTRACT:

The BRANCH DECISION ENCODING SCHEME shown herein overcomes the limitations of a dedicated debug port on a single chip computer processor. A dedicated debug port resolves many of the problems associated with an add-on logic analyzer, except for its limitation of an eight bit data interface. The 8 bit port is required as a trade-off between the device I/O requirements and development tools. During real time program development, it is virtually impossible to monitor the 24 bit program counter through a port only a third as wide. The present invention solves this problem by taking advantage of the sequential characteristics of application programs. There is a discontinuity in the program counter in only a limited number of situations: branches, jumps, subroutine calls and returns from subroutines, exceptions and returns from exceptions, traps and return from traps, and loopbacks to the tops of loops. Therefore, by monitoring most of these discontinuities, it becomes possible to compress the 24 bit data so that it can be output through the 8 bit port in real time. For example, branch decisions can be defined by noting only that they occurred, and if they were taken or not taken, i.e., "1" or "0". The results of the decision tree are contained within the program and can be reconstructed in an external work station. To maintain synchronization, the absolute value of the program counter is updated periodically as the opportunity presents itself. It is always updated following indirect jumps.

5 Claims, 4 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 4

---------- KWIC ---------

Brief Summary Text - BSTX (3):

Heretofore, logic analyzers have been the primary tool used when **trouble shooting** digital computer hardware. However, with the advent of highly pipe-lined, single-chip processors having wide data and address buses, it becomes impractical to bring all necessary buses to the device's **interface** for hardware and program **trouble shooting**. Moreover, it is not possible to multiplex all of the data that one might find useful onto an output bus for this purpose when **debugging** in a real time program environment.

US-PAT-NO: 4959772

DOCUMENT-IDENTIFIER: US 4959772 A

TITLE: System for monitoring and capturing bus data in a computer

DATE-ISSUED: September 25, 1990

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Smith; Royston L.	Plantation	FL	N/A	N/A
Rabaza; Maria V.	Miami Lakes	FL	N/A	N/A

US-CL-CURRENT: 714/48, 714/30

ABSTRACT:

A system for troubleshooting or debugging the various modules of a digital computer which are connected to the ports of a common system bus. A control unit transmits serial function and data codes to a serial bus link gate array whose parallel outputs are further processed to generate steering and clock commands. A first set of bus interface multiplexers are connected to each of the lines of the address, data and control fields of the system bus. A second set of multiplexers cause information represented by digital signals on selected ones of the lines in selected ones of the fields to be stored in a plurality of RAMs in response to the steering commands. A trigger logic is provided for selectively starting and stopping different clock signals in response to the clock commands. Information can be stored in the RAMs continuously until the RAMs are full, until a specified smaller amount of information has been stored in the RAMs, or until a specified event occurs, in response to predetermined ones of the function and data codes. The control unit can retrieve and organize the information captured in the RAMs for display to enable a system operator in locating faults in the computer.

24 Claims, 14 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 11

---------- KWIC ---------

Brief Summary Text - BSTX (9):

It is therefore the primary object of the present invention to provide a system useful in **trouble shooting or debugging** the various modules of a digital computer which are connected to the **ports** of a common system bus.

US-PAT-NO: 5452419

DOCUMENT-IDENTIFIER: US 5452419 A

TITLE: Serial communication control system between nodes having predetermined intervals for synchronous communications and mediating asynchronous communications for unused time in the predetermined intervals

DATE-ISSUED: September 19, 1995

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Di Giulio; Peter C.	Fairfield	CT	N/A	N/A
Lee; David K.	Monroe	CT	N/A	N/A
Riley; David W.	Easton	CT	N/A	N/A
Ryan, Jr.; Frederick W.	N. Haven	CT	N/A	N/A

US-CL-CURRENT: 709/200, 700/56 , 710/1

ABSTRACT:

A cost-effective motion control system communication architecture is provided that supports a centralized control node, distributed control nodes, and smart I/O peripheral control nodes. Networks designed using this architecture, which employs a serial bus, may be readily modified or expanded. The architecture supports both real-time highly periodic communications and event-driven peer-to-peer communications.

17 Claims, 42 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 21

---------- KWIC ---------

Detailed Description Text - DETX (162):

Support is provided for PCNs 220 and DCNs 310 to incorporate boundary-scan as an option which is directly tied to communication. Boundary-scan facilitates automatic testing of electronic boards during manufacturing or product servicing. Boundary-scan accepts a test bit pattern as an input and generates an output bit pattern. Those patterns are predefined when an electronic board is developed. The boundary-scan must be performed while the application **hardware is in a static state**. This will require that the processing unit of DCN 310 or PCN 210 be held in an idle state and may also require that this and other application hardware of DCN 310 and PCN 210 to be held in a reset state as well.

Claims Text - CLTX (54):

means for accepting said node identification signals, said test vectors, and aid scan mode signals, said central control node subsequently setting those of said peripheral control nodes and said distributed control nodes to be tested into a **static hardware state**;

US-PAT-NO: 5499374

DOCUMENT-IDENTIFIER: US 5499374 A

TITLE: Event driven communication network

DATE-ISSUED: March 12, 1996

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Di Giulio; Peter C.	Fairfield	CT	N/A	N/A
Lee; David K.	Monroe	CT	N/A	N/A
Riley; David W.	Easton	CT	N/A	N/A
Ryan, Jr.; Frederick W.	New Haven	CT	N/A	N/A

US-CL-CURRENT: 710/240, 710/107

ABSTRACT:

A cost-effective motion control system communication architecture is provided that supports a centralized control node, distributed control nodes, and smart I/O peripheral control nodes. Networks designed using this architecture, which employs a serial bus, may be readily modified or expanded. The architecture supports both real-time highly periodic communications and event-driven peer-to-peer communications.

20 Claims, 42 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 21

---------- KWIC ---------

Detailed Description Text - DETX (136):

Support is provided for PCNs 220 and DCNs 310 to incorporate boundary-scan as an option which is directly tied to communication. Boundary-scan facilitates automatic testing of electronic boards during manufacturing or product servicing. Boundary-scan accepts a test bit pattern as an input and generates an output bit pattern. Those patterns are predefined when an electronic board is developed. The boundary-scan must be performed while the application **hardware is in a static state**. This will require that the processing unit of DCN 310 or PCN 210 be held in an idle state and may also require that this and other application hardware of DCN 310 and PCN 210 to be held in a reset state as well.

US-PAT-NO: 5812562

DOCUMENT-IDENTIFIER: US 5812562 A

TITLE: Low cost emulation scheme implemented via clock control using JTAG controller in a scan environment

DATE-ISSUED: September 22, 1998

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Baeg; Sanghyeon	Cupertino	CA	N/A	N/A

US-CL-CURRENT: 714/726, 714/731

ABSTRACT:

An integrated circuit includes a test port operating at a first clock speed and at least one functional logic block operating at a second clock speed which is faster than the first clock speed. A control register, formed by boundary scan cells, operates at the first clock speed and provides control inputs to the functional block to control emulation tasks. The control register is writable via a serial shift operation through the test port while the functional block is operating. The outputs of the control register which are being provided to the functional block are held constant during the serial shift operation. An observation register, formed by boundary scan cells, operates at the first clock speed, and is readable through the test port via a serial shift operation, while the functional block is operating. The observation register receives signals from the functional block indicating status of the functional block.

5 Claims, 9 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 6

---------- KWIC ---------

Brief Summary Text - BSTX (5):

The **JTAG** test port (Test Access Port or TAP) has been utilized as a means for emulating and debugging VLSI designs. That is possible because **JTAG** provides control and observability to the core logic portions of the integrated circuit through scan access. However, some prior art emulation approaches using **JTAG** have utilized a complex interface between core logic and the **JTAG** port. That is in part because the **JTAG** port typically operates at a test clock frequency which can be an order of magnitude slower than the system clocks provided to the functional logic within the integrated circuit. For example, U.S. Pat. No. 5,535,331 describes one such relatively complex interface scheme between the **JTAG** controller and functional logic contained within the integrated circuit. Additionally, in many **JTAG** applications, it is necessary to **stop the system clocks** in order to obtain status information about the functional blocks. It would be desirable to have an expandable and flexible interface between the **JTAG** port and the core logic blocks on VLSI chips that is simple, does not require extensive redesign for each new application and provides access to status of the core logic blocks without having to **stop the system clocks** each time.

Detailed Description Text - DETX (6):

By controlling the system clocks using **JTAG,** the functions required for chip emulation can be implemented relatively easy with a minimum cost of design time and silicon penalty. The control necessary includes bypassing (i.e. **stopping) free running system clocks,** clocking for single or multiple cycles, and enabling free running clocks to go back to normal operation. The communication between **JTAG** controller and other functional blocks can be made using two special registers, which resolve the asynchronous relationship and speed gap between TCK and system clocks. In the embodiment described herein, the TCK operates at 10 MHz and while the fastest system clocks operate at 80 MHz.

Detailed Description Text - DETX (9):

Generally, to use the **JTAG** port for debug and emulation purposes, the **system clocks are stopped and the JTAG** controller can scan data in/out to/from all scan flip-flops and set a program counter, set next break address/data break points, and many other control registers to control the design. If a DSP or other internal processor implements address/data break points, instruction stepping is possible. Otherwise, cycle based stepping is supported. Generating single/multiple clocks are implemented using a counter in the clock control logic 205 which uses the count value contained in bits 1-10 of the MCR.

Detailed Description Text - DETX (13):

Once all necessary states have been observed, all **system clocks can be stopped**. In order to scan internal registers, it is necessary to **stop system clocks** going to the registers desired to be scanned. In implementations providing the capability, one can selectively stop the clocks depending on how the values are set up in the MCR. If selective **stopping is not supported, all system clocks will be stopped** as presently implemented in the MSP. A block with system clocks active can not be scanned. The clock stop signal is being issued while the MSP is running with system clocks. Any of the four instructions, MCR/BIST1, MCR/BIST2, MCR/BIST3, and MCR/BIST4 can be used to issue the clock stop signal. MCR/BIST1 and MCR/BIST2 can issue the signals while boundary scan cells are in transparent mode, i.e., input signals external to the chip are entering the chip. The latter two instructions can issue the clock stop signals while all input signals are blocked. See Tables 4-11 in Appendix A for additional details on **JTAG** instructions executed by TAP 111.

Detailed Description Text - DETX (14):

Subsequently, a new MCR load operation takes place to **stop system clocks**. Thus, an instruction is sent to the **JTAG** port selecting the MCR as the data register and data is serially shifted in to the MCR to assert bit 11, sys.sub.-- clk.sub.-- bypass, which will **stop the system clocks** synchronously as described in greater detail in "CLOCK GENERATION FOR TESTING OF INTEGRATED CIRCUITS", by Baeg and Yu, discussed previously. Then the MCR can be loaded with another instruction to determine, e.g., whether the clocks are to be cycled at system speed or to use pseudo system clocks (system clock at TCK speed). As discussed, a 10 bit cycle count (MCR bits 1-10) can be specified. The cnt.sub.-- start bit in the MCR will cause the clock control logic to generate the number of clocks specified in the clock count bits of the MCR. The MCR can control the clock control logic so that the cycle count is executed with regular system clocks or with a clock derived from TCK.

US-PAT-NO: 5793776

DOCUMENT-IDENTIFIER: US 5793776 A

TITLE: Structure and method for SDRAM dynamic self refresh entry and exit using JTAG

DATE-ISSUED: August 11, 1998

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Qureshi; Amjad	San Jose	CA	N/A	N/A
Baeg; Sanghyeon	Cupertino	CA	N/A	N/A

US-CL-CURRENT: 714/724, 714/42

ABSTRACT:

JTAG test logic and a memory controller place an SDRAM in a self refresh mode prior to beginning JTAG testing. The memory controller can complete a current memory access and otherwise prepare for the JTAG test. During the JTAG test, self refresh mode operation of the SDRAM retains data without the need for a clock signal or refresh signals which are suspended for the JTAG test. Accordingly, after the JTAG test, circuit operation can continue without reinitializing data in the SDRAM.

16 Claims, 7 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 6

---------- KWIC ---------

Brief Summary Text - BSTX (11):

A **JTAG** Logic is used in accordance with this invention to asynchronously communicate with a Memory Controller Unit to allow the **stopping of system clocks** while preserving the contents of SDRAM using the self refresh mode. The Memory Controller Unit does not permit the **system clocks to be stopped by the JTAG** Logic for testing until the Memory Controller Unit has finished the current memory access operation. Prior to the **stopping of the system clocks,** the Memory Controller Unit places the SDRAM into self refresh mode to preserve the memory contents.

Brief Summary Text - BSTX (13):

In one embodiment in accordance with this invention, when the **JTAG** Controller wants to **stop the system clock** to allow testing to commence, a signal jtag.sub.-- clk.sub.-- stop.sub.-- req high is asserted and communicated via a Memory Control Register to a Self-Refresh State Machine which is part of the Memory Controller Unit. The Self-Refresh State Machine asserts the signal jtag.sub.-- clk.sub.-- stop.sub.-- request high to a Memory Controller State Machine which finishes the current memory access operation before asserting a signal mcu.sub.-- idle high back to the Self-Refresh State Machine. On assertion of the signal mcu.sub.-- idle high by the Memory Controller State Machine and the presence of signal jtag.sub.-- clk.sub.-- stop.sub.-- req high, the Self-Refresh State Machine places the SDRAM into self refresh mode. The

Self-Refresh State Machine also asserts the signal mcu.sub.-- idle high to an Observation Control Register which is continually scanned by the **JTAG** Controller. If the **JTAG** Controller detects the signal mcu.sub.-- idle high and the signal jtag.sub.-- clk.sub.-- stop.sub.13 req high, a signal sys.sub.13 clk.sub.-- bypass high is asserted by the **JTAG** Controller. The signal sys.sub.-- clk.sub.-- bypass high is asserted to a System Clock Generator Block via the Memory Control Register and causes the system clock to be bypassed.

US-PAT-NO: 4959772

DOCUMENT-IDENTIFIER: US 4959772 A

TITLE: System for monitoring and capturing bus data in a computer

DATE-ISSUED: September 25, 1990

INVENTOR-INFORMATION:

NAME	CITY	STATE	ZIP CODE	COUNTRY
Smith; Royston L.	Plantation	FL	N/A	N/A
Rabaza; Maria V.	Miami Lakes	FL	N/A	N/A

US-CL-CURRENT: 714/48, 714/30

ABSTRACT:

A system for troubleshooting or debugging the various modules of a digital computer which are connected to the ports of a common system bus. A control unit transmits serial function and data codes to a serial bus link gate array whose parallel outputs are further processed to generate steering and clock commands. A first set of bus interface multiplexers are connected to each of the lines of the address, data and control fields of the system bus. A second set of multiplexers cause information represented by digital signals on selected ones of the lines in selected ones of the fields to be stored in a plurality of RAMs in response to the steering commands. A trigger logic is provided for selectively starting and stopping different clock signals in response to the clock commands. Information can be stored in the RAMs continuously until the RAMs are full, until a specified smaller amount of information has been stored in the RAMs, or until a specified event occurs, in response to predetermined ones of the function and data codes. The control unit can retrieve and organize the information captured in the RAMs for display to enable a system operator in locating faults in the computer.

24 Claims, 14 Drawing figures

Exemplary Claim Number: 1

Number of Drawing Sheets: 11

---------- KWIC ---------

Brief Summary Text - BSTX (9):

It is therefore the primary object of the present invention to provide a system useful in **trouble shooting or debugging** the various modules of a digital computer which are connected to the ports of a common system bus.

Detailed Description Text - DETX (12):

The primary function of the bus data monitoring and capturing system (BSS 26 in FIG. 1) is troubleshooting or debugging of the nucleus modules of the computer system which are connected to the ports on the system bus 20. In the preferred embodiment the BSS 26 can monitor one-hundred and forty of the two-hundred and seventy-four system bus signals at any one time upon selection of one of four possible formats. The BSS 26 can capture system bus data continuously until its data buffer is full, until a specified smaller amount of

data has been captured, or until a preselected event occurs. Data capture can also be triggered by predetermined events. The BSS 26 can issue a break-point generated sync pulse for an oscilloscope, as well as **stopping or single-stepping the system clocks** to aid in debugging. The BSS 26 also continuously checks system bus parity and can generate a parity error signal for the memory address field, data field, and expanded data field, on the cycle immediately following the detection of an actual parity error. Backplane signals can also be monitored by the BSS 26.